INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS
Int. J. Sat. Commun., 15, 197–203 (1997)

TIME SLOT ASSIGNMENT IN A HETEROGENEOUS ENVIRONMENT OF A SS/TDMA SYSTEM

SOO-HYUN KIM*[1] AND SEHUN KIM[2]
[1]*Information Infrastructure planning section, Electronics and Telecommunications Research Institute, 161, Kajong-Dong, Yusong-Gu, Taejon 305-350, Korea*
[2]*Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, 373-1, Kusong-Dong, Yusong-Gu, Taejon 305-701, Korea*

SUMMARY

In this paper we consider the time slot assignment for a heterogeneous environment in which circuit-switched traffic and packet-switched traffic share the same satellite. In constructing a single time division multiple access (TDMA) frame for both traffic types, their different characteristics must be taken into account. This problem is known to be NP-complete and a couple of heuristic partial optimization algorithms have been developed. In this paper, we first provide a theoretical result to improve the existing partial optimization algorithms; then a fully optimizing heuristic algorithm is presented. Simulation results show that our algorithm provides a much better solution quality than existing ones. © 1997 John Wiley & Sons, Ltd.

KEY WORDS: SS/TDMA; time slot assignment; heterogeneous environment

1. INTRODUCTION

As the demands of satellite communication rapidly grow, the natural resources they use, the RF spectrum and the geosynchronous orbit, are becoming highly crowded. The utilization of the above natural resources is optimized by employing multibeam antennas and satellite-switched time-division multiple-access (SS/TDMA) techniques. In an SS/TDMA system, a satellite has a number of spot beam antennas, each providing coverage for a limited geographical zone. The solid-state RF switch on-board the satellite provides connections between the various uplink and downlink beams according to the TDMA frame. The frame is divided into time slots. Each time slot represents a particular switching matrix configuration, which transmits a certain number of packets between the connected uplink and downlink beams without conflict. The objective is to permit the transmission of a given traffic load, and to accomplish this with maximum transponder utilization.

Most previous studies on SS/TDMA time slot assignment[1–7,8,9] have focused on a homogeneous traffic environment, consisting of either slowly varying circuit-switched traffic or rapidly varying packet-switched traffic. Recently, Bonuccelli, *et al.*[10] examined the time slot assignment for a heterogeneous environment in which circuit- and packet-switched traffic share the same satellite. Clearly, circuit- and packet-switched traffic have vastly different characteristics and requirements. Circuit-switched traffic (static traffic) is usually composed of a continuous bit-stream of a predetermined bit-rate and requires real-time delivery. Packet-switched traffic (dynamic traffic) is bursty in nature and can be delayed. In constructing a single TDMA frame for both traffic types these differences must be taken into account. Bonuccelli, *et al.*[10] proved that this problem is NP-complete.

In the system studied, the switchpoint closures in each frame slot are determined in response to the changing demand for each frame. A low-capacity order-wire facility (see Reference 5) is assumed for sending connection requests to the satellite. In one possible procedure, each earth station within a zone transmits to the satellite the number of requests for connection to each zone for circuits and packets. For each zone, the onboard scheduler calculates the circuit demand for each path and forms a circuit-switched traffic matrix. It also calculates the new and previously queued packet demand for each path, and forms a packet-switched traffic matrix. For each frame the scheduler determines a switch assignment for each frame slot, first for the circuits, and then for the packets. It then allocates particular stations to the assignments and broadcasts the schedule to the ground.

Bonuccelli, *et al.*[10] suggested a partial optimization approach for this problem. The algorithm in Reference 10 starts with a given TDMA frame for the transmission of the static circuit-switched traffic. As switching conflicts usually prevent full utilization, it is likely that there will be some unused slots in the frame. The algorithm attempts to fill

Correspondence to Soo-Hyun Kim, Information Infrastructure planning section, Electronics and Telecommunications Research Institute, 161, Kajong-Dong, Yusong-Gu, Taejon 305-350, Korea. Email: sookim@smol5.kaist.ac.kr

CCC 0737–2884/97/050197–7 $17.50

Received March 1995
Revised September 1995

these empty slots with packets. The object is to maximize the utilization of these empty slots. This was an excellent idea, but the simulation result was not sufficient because the slots did not optimize fully.

In this paper, we will show that it is possible to generate a TDMA frame for the static circuit-switched traffic more widely, so that more packet-switched traffic can be scheduled. Based on this observation, we can immediately improve the algorithm studied in Reference 10. We suggest a new fully optimizing algorithm. Unlike the algorithm by Bonuccelli, *et al.*,[10] our algorithm generates switching matrices for the static traffic and the dynamic traffic interactively. Simulation results show that our algorithm generates much better solutions than that of Reference 10.

This paper is organized in the following way. The problem formulation is given in Section 2. Section 3 discusses the partial optimization approach. Section 4 presents our fully optimizing algorithm. Section 5 reports and compares the extensive simulation test results of our algorithm with those of the previous algorithm and Section 6 concludes.

2. NOTATIONS AND PROBLEM FORMULATION

The total number of zones in the system is N and we shall label the zones from 1 to N. A traffic matrix R is an $N \times N$ matrix with non-negative integer entries. Entry r_{ij} in R represents the traffic requirement to be transmitted from zone i to zone j. A line of a matrix is one column or row of that matrix, and the line sum is the sum of all entries on one line. The total amount of traffic in R, $\Sigma_{i,j} r_{ij}$, is denoted by $|R|$.

A switching matrix is a traffic matrix with one non-zero entry at most in any line. Thus, a switching matrix represents traffic that can be transmitted without conflict. The length or duration of a switching matrix Q, $L(Q)$ is the magnitude of the largest entry in Q. The transmission of Q takes $L(Q)$ consecutive time slots. A line of switching matrix Q is exposed if all entries in that line are zero. A line with one non-zero entry is called a covered line.

A transmission schedule (or time slot assignment) of a traffic matrix R is a decomposition of R into several $N \times N$ switching matrices $Q_1, Q_2, \ldots, Q_m$, such that $R = Q_1 + Q_2 + \ldots + Q_m$. The total length or transmission time of a schedule is the sum of the lengths of its switching matrices: $L = \Sigma_{k=1}^{m} L(Q_k)$. A schedule for a traffic matrix R is optimal if it achieves maximum transponder utilization among all possible schedules for R. It has been shown in References 3 and 4 that this is equivalent to minimize the schedule length, L.

The time slot assignment problem studied in this paper can be stated as follows. Given $N \times N$ circuit-switched traffic matrix S and packet-switched traffic matrix D, construct schedules $S_1, S_2, \ldots, S_m$ and $D_1, D_2, \ldots, D_m$, such that the amount of scheduled dynamic traffic $|\Sigma_{k=1}^{m} D_k|$ is maximized while the following equations are satisfied:

$$S_k + D_k \text{ is a switching matrix},$$

$$\text{for } k = 1, \ldots, m, \tag{1}$$

$$L(S_k + D_k) = L(S_k), \text{ for } k = 1, \ldots, m, \tag{2}$$

$$\sum_{k=1}^{m} S_k = S, \tag{3}$$

$$\sum_{k=1}^{m} L(S_k) = T_{\max}, \tag{4}$$

$$\sum_{k=1}^{m} D_k \leq D, \tag{5}$$

where $T_{\max}$ is the largest line sum of S, and m is the number of switching matrices. If $T_{\max}$ is smaller than the capacity of one frame, the capacity which is not designated for circuit-switched traffic is reserved for packet-switched traffic only.

We shall refer to matrix D as the dynamic traffic matrix, to D_k as the kth dynamic switching matrix, to S as the static traffic matrix and to S_k as the kth static switching matrix. Equations (1) and (2) ensure that the dynamic traffic is scheduled without any change to the static traffic. Equations (3) and (4) imply that the static traffic is scheduled optimally. Equation (5) ensures that the total amount of dynamic traffic scheduled does not exceed the traffic requirement.

We first derive an upper bound, UB, on the maximum amount of dynamic traffic that can be scheduled. Let R_i and C_j be the line sums of row i and column j, respectively, of the static traffic matrix S. Then $T_{\max} - R_i$ and $T_{\max} - C_j$ are the total number of empty slots in the ith row and jth column, respectively, of the static switching matrices. Similarly, let DR_i and DC_j be the line sums of row i and column j, respectively, of dynamic traffic matrix D. The following UB then becomes an upper bound on the maximum amount of dynamic traffic that can be scheduled:

$$UB = \min\left\{ \sum_{i=1}^{N} \min(T_{\max} - R_i, DR_i), \; \sum_{j=1}^{N} \min(T_{\max} - C_j, DC_j) \right\}.$$

Simulation tests show that this upper bound is better than that in Reference 10.

3. PARTIAL OPTIMIZATION APPROACH

Bonuccelli, *et al.*[10] suggested partial optimization algorithms for this problem which optimize the

scheduling for D only. These algorithms start with a given TDMA frame for the static traffic. Then they assign as many packets as possible to empty slots of the frame. Among their algorithms, the local-optimal algorithm is more practical in the sense that it has relatively low complexity and provides relatively good solutions.

The partial optimization approach provides a solution quickly. However, its solution quality is not guaranteed. We will provide a theorem which shows a way of improving the quality of the partial optimal solution. To prove it, we start with a lemma.

Lemma 1. S and D are switching matrices such that $S+D$ is a switching matrix and $L(S+D)=L(S)$. Suppose that S is decomposed to $\{S^1,\ldots,S^q\}$, such that $S=\Sigma_{l=1}^{q}S^l$ and $L(S)=\Sigma_{l=1}^{q}L(S^l)$. Then there is a decomposition of D $\{D^1,\ldots,D^q\}$, such that $D=\Sigma_{l=1}^{q}D^l$, S^l+D^l is a switching matrix and $L(S^l)=L(S^l+D^l)$ for $l=1,\ldots,q$.

Proof. Let us consider the case where $q=2$. Suppose that $\{D^1,D^2\}$ is a decomposition of D, such that $D=D^1+D^2$ and D^1 and D^2 are switching matrices. Since $S+D$ is a switching matrix, S^l+D^l is also a switching matrix for $l=1,2$. Hence $L(S^l+D^l)=\max\{L(S^l),L(D^l)\}$ for $l=1,2$. Now let us construct a decomposition as follows: $d_{ij}^1=\min\{d_{ij},L(S^1)\}$ and

$$d_{ij}^2=d_{ij}-d_{ij}^1=\begin{cases}d_{ij}-L(S^1)\,, & \text{if } d_{ij}>L(S^1)\,,\\ 0\,, & \text{otherwise}\,,\end{cases}$$

where d_{ij},d_{ij}^l are the (i,j)th elements of matrices D and D^l, respectively. Then $L(D^1)\leq L(S^1)$. And $L(D^2)$ is given by

$$L(D^2)=\begin{cases}L(D)-L(S^1)\,, & \text{if } L(D)>L(S^1)\,,\\ 0\,, & \text{otherwise}\,.\end{cases}$$

Since $L(D)\leq L(S)$, $L(D^2)\leq L(S)-L(S^1)=L(S^2)$. Hence $L(S^l+D^l)=\max\{L(S^l),L(D^l)\}=L(S^l)$ for $l=1,2$. This completes the proof for $q=2$. The proof for $q>2$ immediately follows by successively applying the result for $q=2$.

Theorem 1. Let $\{S_1,\ldots,S_m\}$ be given switching matrices for the static traffic. Suppose that each S_k is decomposed to switching matrices $\{S_k^1,\ldots,S_k^{q_k}\}$ such that

(a) $S_k=\Sigma_{l=1}^{q_k}S_k^l$,
(b) $L(S_k)=\Sigma_{l=1}^{q_k}L(S_k^l)$.

Let $\{D_1,\ldots,D_m\}$ and $\{D_1^1,\ldots,D_1^{q_1},\ldots,D_m^1,\ldots,D_m^{q_m}\}$ be optimal schedules of the dynamic traffic for given $\{S_1,\ldots,S_m\}$ and $\{S_1^1,\ldots,S_1^{q_1},\ldots,S_m^1,\ldots,S_m^{q_m}\}$, respectively. Then

$$|\sum_{k=1}^{m}\sum_{l=1}^{q_k}D_k^l|\geq|\sum_{k=1}^{m}D_k|\,.$$

Proof. From Lemma 1, there exists a decomposition of D_k, $\{\bar{D}_k^1,\ldots,\bar{D}_k^{q_k}\}$, such that $D_k=\Sigma_{l=1}^{q_k}\bar{D}_k^l$, $S_k^l+\bar{D}_k^l$ is a switching matrix and $L(S_k^l)=L(S_k^l+\bar{D}_k^l)$ for $l=1,\ldots,q_k$. The schedules $\{S_1^1,\ldots,S_1^{q_1},\ldots,S_m^1,\ldots,S_m^{q_m}\}$ and $\{\bar{D}_1^1,\ldots,\bar{D}_1^{q_1},\ldots,\bar{D}_m^1,\ldots,\bar{D}_m^{q_m}\}$ therefore satisfy Equations (1)–(5). Since for given $\{S_1^1,\ldots,S_1^{q_1},\ldots,S_m^1,\ldots,S_m^{q_m}\}$, $\{D_1^1,\ldots,D_1^{q_1},\ldots,D_m^1,\ldots,D_m^{q_m}\}$ is an optimal schedule among all possible schedules for dynamic traffic, satisfying Equations (1)–(5),

$$|\sum_{k=1}^{m}\sum_{l=1}^{q_k}D_k^l|\geq|\sum_{k=1}^{m}\sum_{l=1}^{q_k}\bar{D}_k^l|=|\sum_{k=1}^{m}D_k|\,.$$

This theorem implies that if we apply a partial optimization algorithm after decomposing the switching matrix of the static traffic, we can obtain a better solution for the dynamic traffic. For example, Figure 1 is the example presented in Reference 10. By applying the local-optimal algorithm to this example, we obtain a schedule where the total number of packets assigned is 21. But, this number can be increased by first decomposing the traffic matrix S into five switching matrices $S_1,\ldots,S_5$, as in Figure 2. The algorithm in Section 4 contains the method for decomposition. Notice that the length of the schedule is unchanged. Figure 2 also shows the result obtained by applying the local-optimal algorithm starting with the TDMA frame $\{S_1,\ldots,S_5\}$. Notice that the total number of packets assigned is increased from 21 to 33. The upper bound derived in the previous section for this example is 33. Hence the schedule in Figure 2 is actually a true optimal solution. In this case, decomposition and a partial optimization approach generates a fully optimal solution. However, in general this is not the case.

4. FULL OPTIMIZATION ALGORITHM

In this section we shall propose a full optimization algorithm which produces the static and dynamic switching matrices interactively. This algorithm generates switching matrices for the static traffic while considering the possibility of inserting additional dynamic traffic to the generated static switching matrix.

We first construct a $2N\times 2N$ static traffic matrix

						5	
4							
	7						
							2
				5			

4		2		5			
		1	7			3	2
6			3		8		
	9		1	4			4
	6	3			5	10	
8		12	3		2		
3				9		5	
		6	1		4		7

Figure 1. An example of matrix S and matrix D

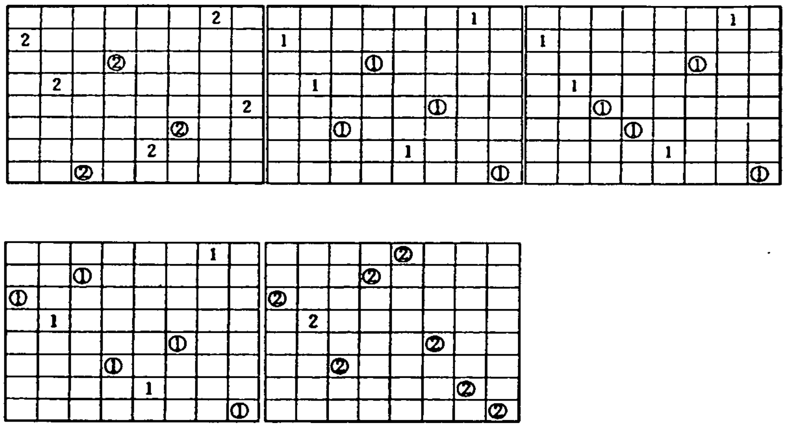

Figure 2. A schedule for the example in Figure 1: the entries of D are encircled. The amount of dynamic traffic scheduled is 33

$\bar{S}$ from S, such that the total traffic in each line of $\bar{S}$ is equal to T_{max}, by augmenting the dummy traffic matrices X,Y,Z to S as follows:

$$\bar{S} = \begin{pmatrix} S & X \\ Y & Z \end{pmatrix},$$

where X,Y are diagonal matrices. Elements in X,Y are determined by the condition that the total traffic in each line of $\bar{S}$ is equal to T_{max}. Let Z be a non-negative matrix such that the total traffic in each line of $\bar{S}$ is equal to T_{max}. By augmenting the dummy traffic matrices, the total static traffic in each line of $\bar{S}$ is equal to T_{max}, resulting in a total static traffic to be scheduled $2NT_{max}$. It is known that an optimal TDMA frame for $\bar{S}$ with a length equal to T_{max} can be easily generated.[9]

The matrix $\bar{S}$ can be represented as a bipartite graph $G_{\bar{S}}(U,V,E)$, where the nodes in U correspond to the rows of $\bar{S}$, and the nodes in V to the columns. The edge set E contains an edge joining the nodes u_i and v_j, denoted (u_i,v_j), if and only if, $\bar{s}_{ij} \neq 0$. A numerical value, $\bar{s}_{ij}$, is assigned to each edge. The resulting graph will be called the network associated with $\bar{S}$. For this graph, we find a complete matching. The set of cells in S corresponding to this complete matching, provide a switching matrix S_1 for the static traffic. Suppose that a matched edge (u_i,v_j) corresponds to dummy traffic. This dummy traffic can belong to one of three matrices X,Y,Z. If it belongs to X, then $1 \le i \le N$, $j = N+i$, and row i in S_1 has no non-zero elements, i.e. row i is exposed. The dynamic traffic can be assigned to row i and $\bar{s}_{i,N+i}$ is an upper bound on the maximum schedulable dynamic traffic for row i. Similarly, if it belongs to Y, then $1 \le j \le N$, $i = N+j$, and column j is exposed. The dynamic traffic can be assigned to column j and $\bar{s}_{N+j,j}$ is an upper bound on the maximum schedulable dynamic traffic for column j. If it belongs to Z, then $N+1 \le i, j \le 2N$. Since we are interested in the assignment to rows $1,\ldots,N$ and columns $1,\ldots,N$, we simply neglect this case.

To assign the dynamic traffic, we perform an assignment procedure. Let $DR_i(DC_j)$ be the sum of row i (column j) of matrix D and $w_i^r = \bar{s}_{i,N+i} - DR_i(w_j^c = \bar{s}_{N+j,j} - DC_j)$ for $i,j = 1,2,\ldots,N$. If $w_i^r(w_j^c) \ge 0$, then $w_i^r(w_j^c)$ represents the minimum possible number of unused surplus slots in row i (column j) even after the assignment of dynamic traffic. If $w_i^r(w_j^c) < 0$, then all surplus slots could be used and the minimum possible number of unused surplus slots is zero.

The dynamic traffic with the smallest row and column surplus slots may cause a bottleneck in the scheduling. Hence $w_{ij} = \max\{w_i^r,0\} + \max\{w_j^c,0\}$ is used as the weight for non-zero entry d_{ij} for $i,j = 1,2,\ldots,N$. Since the dynamic traffic can be assigned to the exposed lines, the calculation of w_{ij} is executed only for these exposed lines. Therefore, if both row i and column j are exposed, we calculate w_{ij}. Otherwise, we set a sufficiently large number to w_{ij}, and then solve the weighted matching problem for the dynamic traffic which minimizes $\Sigma_i\Sigma_j w_{ij}$. The cells in D corresponding to a solution of this weighted matching problem are candidates for the switching matrix for dynamic traffic. Among these cells, only the cells corresponding to the exposed rows of S_1 can be scheduled, since a switching matrix can have at most one non-zero element in each line. Let D_1 be a switching matrix for the dynamic traffic constructed in this way.

The matrix S_1 and D_1 are therefore the first set of switching matrices for the static and dynamic traffic, respectively. However, it would be more efficient to truncate these matrices so that all non-zero entries are equal. Let δ_1 be the smallest non-zero value of cells in $\bar{S}$, corresponding to the set of matched edges. Similarly, let δ_2 be the smallest non-zero value of cells in D, corresponding to the set of weighted matched edges. Let $\delta = \min\{\delta_1,\delta_2\}$. Then truncate all non-zero elements in S_1 and D_1 to the value δ. This completes the first iteration for the generation of S_1 and D_1. Subtract S_1 and D_1 from S and D, respectively, and then repeat. We

can easily show that the switching matrices constructed in this way satisfy Equations (1)–(5).

Algorithm Time slot assignment for heterogeneous environment (TSAH)

Step 0: Initialization
Let $S=\{s_{ij}\}$ and $D=\{d_{ij}\}$ be given traffic matrices; set $k \leftarrow 1$;

Step 1:
1.0. $(T_{\max})$
$SR_i = \Sigma_{j=1}^{N} s_{ij},\ i = 1,\ldots,N;$
$SC_j = \Sigma_{i=1}^{N} s_{ij},\ j = 1,\ldots,N;$
$T_{\max} = \max_{i,j}\{SR_i, SC_j\};$

1.1. Create augmented traffic matrix
$\bar{S} \leftarrow 0;$
$\bar{s}_{ij} \leftarrow s_{ij},\ i,j = 1,\ldots,N;$
$\bar{s}_{i,N+i} \leftarrow T_{\max} - SR_i,\ i = 1,\ldots,N;$
$\bar{s}_{N+j,j} \leftarrow T_{\max} - SC_j,\ j = 1,\ldots,N;$
For $i = 1,\ldots,N$ and $j = 1,\ldots,N$,

if $SR_{N+i}\ (= \Sigma_{j=1}^{2N} \bar{s}_{N+i,j}) \geq SC_{N+j}\ (= \Sigma_{i=1}^{2N} \bar{s}_{i,N+j})$ then $\bar{s}_{N+i,N+j} \leftarrow T_{\max} - SR_{N+i}$ else $\bar{s}_{N+i,N+j} \leftarrow T_{\max} - SC_{N+j};$

Step 2: (Form switching matrices)
2.0. $S_k \leftarrow 0;\ D_k \leftarrow 0;$
2.1. Set up the network associated with $\bar{S}$;
2.2. Find a complete matching M in the network associated with $\bar{S}$;

$M = \{(u_1,m(u_1)),(u_2,m(u_2)),\ldots,(u_{2N},m(u_{2N}))\}$, where $m(u_i)$ is the mate of u_i;
$R \leftarrow \varnothing;\ C \leftarrow \varnothing;$
For $u_i = 1,\ldots,N$, if $m(u_i) > N$ then $R \leftarrow R + \{u_i\}$;
For $u_i = N+1,\ldots,2N$, if $m(u_i) \leq N$ then $C \leftarrow C + \{u_i - N\}$;

2.3. Create an $N \times N$ weight matrix $W=\{w_{ij}\}$ as follows:
For $i \in R$, if $DR_i\ (= \Sigma_{j=1}^{N} d_{ij}) > 0$ then $w_i^r = \bar{s}_{i,N+i} - DR_i$; if $w_i^r < 0$ then $w_i^r \leftarrow 0$; For $j \in C$, if $DC_j (= \Sigma_{i=1}^{N} d_{ij}) > 0$ then $w_j^c = \bar{s}_{N+j,j} - DC_j$; if $w_j^c < 0$ then $w_j^c \leftarrow 0$; For $i,j = 1,\ldots,N$, if $i \in R, j \in C$, and $d_{ij} \neq 0$ then $w_{ij} = w_i^r + w_j^c$, otherwise, $w_{ij} = \infty$; Find a weighted matching A which minimizes $\Sigma_i \Sigma_j w_{ij}$ in the network associated with W;
$A = \{(x_1,a(x_1)),(x_2,a(x_2)),\ldots,(x_N,a(x_N))\}$; where $a(x_i)$ is the mate of x_i;

2.4. For $i = 1,\ldots,N$, if $i \notin R$ then store $s_{i,m(i)}$ in the matrix S_k, else store $d_{i,a(i)}$ in the matrix D_k;

2.5 $\delta_1 = \min\{\bar{s}_{u_1,m(u_1)},\ldots,\bar{s}_{u_{2N},m(u_{2N})}\};$
$\delta_2 = \min\{d_{x_i,a(x_i)} \mid d_{x_i,a(x_i)} > 0,\ i \in \{1,\ldots,N\}\};$
$\delta = \min\{\delta_1,\delta_2\};$
Form a switching matrix $S_k + D_k$ by replacing all its nonzero entries by the value δ;

Step 3: (Updating and termination)
$S \leftarrow S - S_k;\ D \leftarrow D - D_k;$
If S contains no non-zero entry, then STOP. Otherwise, set $k \leftarrow k + 1$; go to Step 1.

The computation of a complete matching in Step 2.2. and the weighted matching computation in Step 2.3. can be carried out in $O(N^3)$ time.[11] Since Step 2 is executed at most $T_{\max}$ times, the overall worst-case time complexity is $O(T_{\max}N^3)$.

5. SIMULATION RESULTS

TSAH is implemented in Pascal and simulation tests have been performed. Traffic matrices of 4×4 and 7×7 are tested. For each size, dense and sparse matrix cases are tested. Dense cases have few null entries and sparse cases have nearly 50 per cent null entries. The static traffic matrix entries are randomly generated from the uniform distribution between $[0,P]$, where P is the largest possible entry in S. The dynamic traffic matrix entries are randomly generated from a Poisson distribution with a mean of 4·5. For each case of matrix size and density, four different values of P are tested. For each test, we generate 200 problems and report the average.

Tables I–IV show the results of our simulation experiment. They present the upper bound obtained in Section 2, the number of dynamic traffic scheduled and the average percentage deviation from the upper bound for each algorithm. In Tables I–IV, LO-GG is the local-optimal algorithm in Reference 10, combined with the algorithm in Reference 8, which generates a TDMA frame for the static traffic. The algorithm in Reference 8 generates switching matrices with equal non-zero entries, which are a quite decomposed TDMA frame. As we have shown in Section 3, the local-optimal algorithm generates a better solution for a decomposed case. If the local-optimal algorithm is combined with other algorithms which generate a TDMA frame with less decomposed switching matrices, the results of the local-optimal algorithm would be much worse.

Tables I–IV show that the solution quality of our algorithm TSAH, is much better than that of the local-optimal algorithm, even though the local-optimal algorithm is combined with a decomposing algorithm in Reference 8. Notice that the percentage deviation is measured from the upper bound, UB, not from the true optimal value. We think the results of TSAH are very close to optimal values.

As the value of P increases, i.e. as the difference between the quantity of static traffic and the quantity of dynamic traffic gets larger, TSAH provides better solutions. Tables I–IV also show that TSAH has good performance for sparse matrices.

6. CONCLUSIONS

In this paper, we considered the time slot assignment for a heterogeneous environment in which circuit-

Table I. Number of dynamic traffic scheduled

4 × 4 Dense Matrices:				
Algorithm	$P = 10$	$P = 20$	$P = 40$	$P = 100$
UB	28	37	44	47
LO-GG	26·04 (9·43)	32·24 (14·66)	37·71 (15·86)	39·32 (17·07)
TSAH	26·65 (7·51)	34·14 (9·93)	41·01 (8·58)	43·38 (8·76)

Numbers in parentheses are the average percentage deviation from the upper bound.

Table II. Number of dynamic traffic scheduled

4 × 4 Sparse Matrices:				
Algorithm	$P = 10$	$P = 20$	$P = 40$	$P = 100$
UB	17	22	24	23
LO-GG	15·94 (11·59)	19·98 (10·28)	21·84 (9·73)	20·71 (12·20)
TSAH	16·65 (7·97)	21·20 (5·44)	23·48 (3·61)	22·58 (4·93)

Numbers in parentheses are the average percentage deviation from the upper bound.

Table III. Number of dynamic traffic scheduled

7 × 7 Dense Matrices:				
Algorithm	$P = 10$	$P = 20$	$P = 40$	$P = 100$
UB	92	132	154	167
LO-GG	85·53 (8·49)	116·41 (12·65)	131·55 (15·23)	142·63 (15·21)
TSAH	86·09 (7·90)	119·71 (10·35)	138·80 (10·63)	151·26 (10·07)

Numbers in parentheses are the average percentage deviation from the upper bound.

Table IV. Number of dynamic traffic scheduled

7 × 7 Sparse Matrices:				
Algorithm	$P = 10$	$P = 20$	$P = 40$	$P = 100$
UB	64	76	84	85
LO-GG	59·30 (9·05)	70·47 (7·91)	77·95 (7·66)	78·52 (8·42)
TSAH	60·41 (7·38)	72·65 (5·15)	81·33 (3·72)	82·48 (4·00)

Numbers in parentheses are the average percentage deviation from the upper bound.

and packet-switched traffic share the same satellite. Our algorithm provides solutions better than the existing algorithm and has relatively low computational complexity.

Our algorithm is also applicable to a case where the traffic is homogeneous in nature and has different priorities. A reasonable approach may be to schedule the high priority traffic as the static traffic and the low priority traffic as the dynamic traffic. Extending our algorithm to a case where the traffic has more than two different priority types would be an interesting study for future research work.

REFERENCES

1. Y. Ito, Y. Urano, T. Muratani and M. Yamaguchi, 'Analysis of a switch matrix for an SS/TDMA system', *Proc. IEEE*, **65**(3), 411–419 (1977).
2. T. Inukai, 'An efficient SS/TDMA time slot assignment algorithm', *IEEE Trans. Com.*, **COM-27**(10), 1449–1455 (1979).
3. G. Bongiovanni, D. Coppersmith and C. K. Wong, 'An optimal time slot assignment algorithm for an SS/TDMA system with variable number of transponders', *IEEE Trans. Com.*, **COM-29**(5), 721–726 (1981).
4. I. S. Gopal, G. Bongiovanni, M. A. Bonuccelli, D. T. Tang and C. K. Wong, 'An optimal switching algorithm for multibeam satellite systems with variable bandwidth beams', *IEEE Trans. Com.*, **COM-30**(11), 2475–2481 (1982).

5. I. S. Gopal, D. Coppersmith and C. K. Wong, 'Minimizing packet waiting time in a multibeam satellite system', *IEEE Trans. Com.*, **COM-30**(2), 305–316 (1982).
6. I. S. Gopal and C. K. Wong, 'Minimizing the number of switchings in an SS/TDMA system', *IEEE Trans. Com.*, COM-33(6), 497–501 (1985).
7. C. A. Pomalaza-Raez, 'A note on efficient SS/TDMA assignment algorithms', *IEEE Trans. Com.*, **COM-36**(9), 1078–1082 (1988).
8. A. Ganz and Y. Gao, 'Efficient algorithms for an SS/TDMA scheduling', *IEEE Trans. Com.*, **COM-40**(8), 1367–1374 (1992).
9. S. Kim and S.-H. Kim, 'An efficient algorithm for generalized SS/TDMA scheduling with satellite cluster and intersatellite links', *Int. J. Sat. Comm.*, 13, 31–37 (1995).
10. M. A. Bonuccelli, I. Gopal and C. K. Wong, 'Incremental time-slot assignment in SS/TDMA satellite systems', *IEEE Trans. Com.*, **COM-39**(7), 1147–1155 (1991).
11. C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, 1982.

Authors' biographies:

Soo-Hyun Kim received his BS in statistics from Korea University and his MS and Ph.D. degrees in management science from the Korea Advanced Institute of Science and Technology (KAIST) in 1991, 1993 and 1997, respectively. Currently, he is a post-doctoral researcher at Electronics and Telecommunications Research Institute (ETRI). His research interests are in the areas of combinatorial and nonlinear optimization, and optimal design and analysis of communication systems, particularly, satellite communication systems and cellular mobile systems.

Sehun Kim received his BS in physics from Seoul National University, Seoul, Korea, in 1972, and his MS and PhD degrees in operations research from Stanford University in 1978 and 1981, respectively. From 1981 to 1982 he worked at Systems Control, Inc. In 1982 he joined the faculty of the Korea Advanced Institute of Science and Technology (KAIST), where he is currently a Professor of Industrial Engineering. He was a visiting professor at Arizona State University from 1986 to 1987. His research interests are in the areas of application of optimization techniques, to the design and analysis of computer and communication systems. He has published a number of papers in *IEEE Transactions on Vehicular Technology*, *Mathematical Programming*, *Operations Research Letters*, *International Journal of Satellite Communications*, *INFOR*, *Journal of Applied Physics*. He served as Editor of the *Journal of Korean Operations Research Society* from 1989 to 1991.